

L Number	Hits	Search Text	DB	Time stamp
1	3	(( "6,327,582" ) or ( "5,946,674" ) or ("5,343,554" ) ).PN.	USPAT; US-PGPUB	2004/02/21 15:28

## USPTO PATENT FULL-TEXT AND IMAGE DATABASE

[Home](#)
[Quick](#)
[Advanced](#)
[Pat Num](#)
[Help](#)
[Bottom](#)
[View Cart](#)

Searching 1790 to present...

**Results of Search in 1790 to present db for:**

**(("genetic algorithm" AND tree) AND lisp): 28 patents.**
















*Hits 1 through 28 out of 28*














Jump To

Refine Search

"genetic algorithm" and tree and lisp

PAT. NO. Title

- 1 [6,668,312](#)  [System, method, and article of manufacture for dynamically profiling memory transfers in a program](#)
- 2 [6,634,019](#)  [Toggling software characteristics in a fault tolerant and combinatorial software environment system, method and medium](#)
- 3 [6,571,226](#)  [Method and apparatus for automated design of chemical synthesis routes](#)
- 4 [6,564,194](#)  [Method and apparatus for automatic synthesis controllers](#)
- 5 [6,532,453](#)  [Genetic programming problem solver with automatically defined stores loops and recursions](#)
- 6 [6,493,686](#)  [Computer implemented machine learning method and system including specifically defined introns](#)
- 7 [6,449,603](#)  [System and method for combining multiple learning agents to produce a prediction method](#)
- 8 [6,424,959](#)  [Method and apparatus for automatic synthesis, placement and routing of complex structures](#)
- 9 [6,360,191](#)  [Method and apparatus for automated design of complex structures using genetic programming](#)
- 10 [6,327,582](#)  [Method and system for genetic programming](#)
- 11 [6,128,607](#)  [Computer implemented machine learning method and system](#)
- 12 [6,098,059](#)  [Computer implemented machine learning method and system](#)
- 13 [6,088,510](#)  [Computer system and method for generating and mutating objects by iterative evolution](#)
- 14 [6,058,385](#)  [Simultaneous evolution of the architecture of a multi-part program while solving a problem using architecture altering operations](#)
- 15 [5,946,674](#)  [Turing complete computer implemented machine learning method and system](#)

- 16 [5,946,673](#)  [Computer implemented machine learning and control system](#)
  - 17 [5,930,780](#)  [Distributed genetic programming](#)
  - 18 [5,867,397](#)  [Method and apparatus for automated design of complex structures using genetic programming](#)
  - 19 [5,841,947](#)  [Computer implemented machine learning method and system](#)
  - 20 [5,742,738](#)  [Simultaneous evolution of the architecture of a multi-part program to solve a problem using architecture altering operations](#)
  - 21 [5,701,400](#)  [Method and apparatus for applying if-then-else rules to data sets in a relational data base and generating from the results of application of said rules a database of diagnostics linked to said data sets to aid executive analysis of financial data](#)
  - 22 [5,390,282](#)  [Process for problem solving using spontaneously emergent self-replicating and self-improving entities](#)
  - 23 [5,343,554](#)  [Non-linear genetic process for data encoding and for solving problems using automatically defined functions](#)
  - 24 [5,249,259](#)  [Genetic algorithm technique for designing neural networks](#)
  - 25 [5,148,513](#)  [Non-linear genetic process for use with plural co-evolving populations](#)
  - 26 [5,140,530](#)  [Genetic algorithm synthesis of neural networks](#)
  - 27 [5,136,686](#)  [Non-linear genetic algorithms for solving problems by finding a fit composition of functions](#)
  - 28 [4,935,877](#)  [Non-linear genetic algorithms for solving problems](#)
- 

[Top](#)

[View Cart](#)

[Home](#)

[Quick](#)

[Advanced](#)

[Pat Num](#)

[Help](#)



Subscribe Register Login  
(Full Service) (Limited Service, Free)

Search: ☒ The ACM Digital Library ☐ The Guide

"genetic algorithm" and tree and lisp

Feed

Terms used genetic algorithm and tree and lisp

Sort results  
by

relevance

Save results to a Binder

Try

Search Tips

Try

☐ Open results in a new window

Display results

expanded form

Results 1 - 20 of 200

Result page: 1 2 3 4 5 6 7 8 9

Best 200 shown

## 1 Evolutionary learning of graph layout constraints from examples

Toshiyuki Masui

November 1994

Proceedings of the 7th annual ACM symposium on User interface s

Full text available: pdf(586.25 KB)

Additional Information: full citation, abstract, reference

We propose a new evolutionary method of extracting user preferences from example layout system. Using stochastic methods such as simulated annealing and genetic algorithms can find a good layout using an evaluation function which can calculate how good the evaluation function is usually not known beforehand, and it might vary from user to user system several pairs of good and bad examples.

Keywords: adaptive user interface, genetic algorithms, genetic programming, graph layout programming by example

## 2 Artificial evolution for computer graphics

Karl Sims

July 1991 ACM SIGGRAPH Computer Graphics , Proceedings of the 18th annual conference on interactive techniques, Volume 25 Issue 4

Full text available: pdf(8.74 MB)

Additional Information: full citation, references, citing:

### 3 Designing laboratory modules for novices in an undergraduate AI course tr:

Robert M. Aiken, Dean Allemang, Thomas Wehrle

March 1992 ACM SIGCSE Bulletin , Proceedings of the twenty-third SIGCSE technical education, Volume 24 Issue 1

Full text available:  pdf(446.93 KB)

Additional Information: full citation, abstract, references,

A current joint project between three institutions in Switzerland has as its goal t software in teaching principles of AI at the University level. The modules of this illustrate basic concepts of Artificial Intelligence in a uniform and self-contained considerations that were adopted in order to make the presentation of this mate

### 4 Data clustering: a review

A. K. Jain, M. N. Murty, P. J. Flynn

September 1999

ACM Computing Surveys (CSUR), Volume 31 Issue 3

Full text available:  pdf(636.24 KB)

Additional Information: full citation, abstract, references, citir

Clustering is the unsupervised classification of patterns (observations, data item (clusters). The clustering problem has been addressed in many contexts and by reflects its broad appeal and usefulness as one of the steps in exploratory data : problem combinatorially, and differences in assumptions and contexts in differer useful generic co ...

Keywords: cluster analysis, clustering applications, exploratory data analysis, in- unsupervised learning

### 5 Machine learning in the liberal arts curriculum

Clare Bates Congdon

March 2000 ACM SIGCSE Bulletin , Proceedings of the thirty-first SIGCSE technical s education, Volume 32 Issue 1

Full text available:  pdf(461.16 KB)

Additional Information: full citation, abstract, referen

Machine learning is typically considered a graduate-level course with an artificial However, it does not need to be positioned this way, and in the liberal arts curri to offering this course to undergraduate students. An undergraduate course in n structured to introduce research concepts and to work within a research paradig refle ...

## 6 AGENTS: a distributed client-server system for leaf cell generation

Dilvan de Abreu Moreira, Les T. Walczowski

January 1997 ACM Transactions on Design Automation of Electronic Systems (TOD

Full text available:  pdf(727.66 KB)

Additional Information: full citation, abstract, referenc



The AGENTS system is a set of programs designed to generate automatically the BICMOS, and bipolar leaf cells. The system is formed from four server programs: broker. The placer places components in a cell, the router wires the circuits sent information that is dependent upon the fabrication process, such as the design of the other serv ...

Keywords: client/server model, genetic algorithms, software agents

## 7 Book reviews

December 1999

intelligence, Volume 10 Issue 4

Full text available:  pdf(385.82 KB)  html(66.61 KB) Additional Information: full citation, references, index te

## 8 Meta optimization: improving compiler heuristics with machine learning

Mark Stephenson, Saman Amarasinghe, Martin Martin, Una-May O'Reilly

May 2003 ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2003 conference implementation, Volume 38 Issue 5

Full text available:  pdf(302.23 KB)

Additional Information: full citation, abstract, referen


Compiler writers have crafted many heuristics over the years to approximately s a heuristic that performs well on a broad range of applications is a tedious and c Meta Optimization, a methodology for automatically fine-tuning compiler heurist machine-learning techniques to automatically search the space of compiler heur design complexity by relieving c ...

Keywords: compiler heuristics, genetic programming, machine learning, priority

## 9 Efficient handling of multiple inheritance hierarchies

Yves Caseau

October 1993 ACM SIGPLAN Notices , Proceedings of the eighth annual conference o languages, and applications, Volume 28 Issue 10

Full text available:  pdf(1.63 MB)

Additional Information: full citation, references, citing:

## 10 Enhancing information retrieval by automatic acquisition of textual relations

Agneta Bergström, Patricija Jaksetic, Peter Nordin

January 2000

Proceedings of the 5th international conference on Intelligent user interfaces

Full text available:  pdf(633.96 KB)

Additional Information: full citation, abstract, references

We have explored a novel method to find textual relations in electronic document semantic networks. This can be used for enhancing information retrieval and since extraction of relations from text enables easier updating of electronic dictionaries search input and hit output on small screens such as cell phones and PDAs (Persson et al., 1999).



Keywords: genetic programming, information retrieval, machine learning, natural language processing, semantic networks

## 11 Evolving virtual creatures

Karl Sims

July 1994

Proceedings of the 21st annual conference on Computer graphics and interactive techniques

Full text available:  pdf(84.65 KB)  ps(219.40 KB)

Additional Information: full citation, abstract, references


This paper describes a novel system for creating virtual creatures that move and interact in virtual physical worlds. The morphologies of creatures and the neural systems for controlling them are generated automatically using genetic algorithms. Different fitness evaluation functions are used to evolve creatures towards specific behaviors such as swimming, walking, jumping, and so on. The system that uses no ...

## 12 Knowledge-based document retrieval in office environments: the Kabiria system

Augusto Celentano, Maria Grazia Fugini, Silvano Pozzi

July 1995






ACM Transactions on Information Systems (TOIS), Volume 13 Issue 4

Full text available:  pdf(2.14 MB)

Additional Information: full citation, abstract, references, citing papers

In the office environment, the retrieval of documents is performed using the contextual information about the procedural context where the documents are used, and in that discipline the life of documents within a given application domain. To fulfill this need, we propose a document retrieval model and system based on the representation of the semantic contents of documents and the procedural context of their use.


Keywords: browser, class, hypertext, instance, knowledge base, link, object oriented, retrieval, system

- 13 A weighted coding in a genetic algorithm for the degree-constrained minimi**  
Günther R. Raidl, Bryant A. Julstrom  
March 2000 Proceedings of the 2000 ACM symposium on Applied computing  
Full text available:  pdf(499.35 KB) Additional Information: full citation, references, citings, inc
- Keywords: degree-constrained minimum spanning trees, genetic algorithms, we
- 14 Designing telecommunications networks using genetic algorithms and prob**  
Faris N. Abuali, Dale A. Schoenefeld, Roger L. Wainwright  
April 1994 Proceedings of the 1994 ACM symposium on Applied computing  
Full text available:  pdf(567.60 KB) Additional Information: full citation, references, citings
- 15 Solving the three-star tree isomorphism problem using genetic algorithms**  
Faris N. Abuali, Roger L. Wainwright, Dale A. Schoenefeld  
February 1995 Proceedings of the 1995 ACM symposium on Applied computing  
Full text available:  pdf(861.60 KB) Additional Information: full citation, references, citings, index terms
- 16 Session 5: university education: The development and operation of edinbur**  
**summer scholarship programme**  
G. V. Wilson, N. B. MacDonald, C. Thornborrow, C. M. Brough  
November 1994 Proceedings of the 1994 ACM/IEEE conference on Supercomputing  
Full text available:  pdf(944.30 KB) Additional Information: full citation, abstract, ref
- Between 1987 and 1994, more than 100 students in a broad range of disciplines  
Edinburgh Parallel Computing Centre. Many of these students have since taken  
graduate work and industry, and over a quarter of EPCC's technical staff are alu  
describes the evolution and present operation of the Summer Scholarship Progra
- 17 IS '97: model curriculum and guidelines for undergraduate degree program**  
Gordon B. Davis, John T. Gorgone, J. Daniel Couger, David L. Feinstein, Herbert E.  
December 1997 ACM SIGMIS Database , Guidelines for undergraduate degree progr  
for undergraduate degree programs in information systems, Volume  
Full text available:  pdf(7.24 MB) Additional Information: full citation



**18 Context-sensitive interprocedural points-to analysis in the presence of func**

Maryam Emami, Rakesh Ghiya, Laurie J. Hendren

June 1994 ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1994 conference  
implementation, Volume 29 Issue 6Full text available:  pdf(1.74 MB)


Additional Information: full citation, abstract, references, cit

This paper reports on the design, implementation, and empirical results of a new problem in C. The method is based on approximating the points-to relationships can be used to generate alias pairs, or used directly for other analyses and trans context-sensitive interprocedural information based on analysis over invocation including re ...

**19 What have we learnt from using real parallel machines to solve real problem**

G. C. Fox

January 1989 Proceedings of the third conference on Hypercube concurrent comput

Full text available:  pdf(4.08 MB)

Additional Information: full citation, abstract, references, ci

We briefly review some key scientific and parallel processing issues in a selectio parallel machines. We include the MIMD hypercube transputer array, BBN Butte MPP and Connection Machine from Thinking Machines. We use a space-time ana division into synchronous, loosely synchronous and asynchronous problems is he suitable for SIMD or MIMD ...

**20 Strategy game programming projects**

Timothy Huang

April 2001 The Journal of Computing in Small Colleges , Proceedings of the sixth an  
The journal of computing in small colleges, Volume 16 Issue 4Full text available:  pdf(208.83 KB)

Additional Information: full citation, abstract, references,

In this paper, we show how programming projects centered arou of computer players for strategy games can play a meaningful r both in and out of the classroom. We describe several game-rela author in a variety of pedagogical situations, including introduct as independent and collaborative research projects. These proje and develop advanced da ...

Results 1 - 20 of 200

Result page: **1** 2 3 4 5 6 7

The ACM Portal is published by the Association for Computing Machinery. C

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Cont](#)Useful downloads:  Adobe Acrobat  QuickTime  Windows Med



Subscribe Register Login  
(Full Service) (Limited Service, Free)

Search: ☒ The ACM Digital Library ☐ The Guide

"genetic algorithm" and tree and lisp

Feed

Terms used genetic algorithm and tree and lisp

Sort results  
by

relevance

Save results to a Binder

Search Tips

☐ Open results in a new window

Try  
Try

Display results

expanded form

Results 21 - 40 of 200

Result page: previous 1 2 3 4 5 6 7

Best 200 shown

## 21 Minimizing row displacement dispatch tables

Karel Driesen, Urs Hölzle

October 1995 ACM SIGPLAN Notices , Proceedings of the tenth annual conference on languages, and applications, Volume 30 Issue 10

Full text available: pdf(1.81 MB)

Additional Information: full citation, abstract, references, c

Row displacement dispatch tables implement message dispatching for dynamica overhead of one memory indirection plus an equality test. The technique is simil is, however, restricted to statically typed languages like C++. We show how to i tables to approximately the same size as virtual function tables. The scheme is l Experiments on a numbe ...

## 22 Three-phase chip planning &mdash; an improved top-down chip planning s

Bernd Schürmann, Joachim Altmeyer, Gerhard Zimmermann

November 1992 Proceedings of the 1992 IEEE/ACM international conference on Com

Full text available: pdf(980.54 KB)

Additional Information: full citation, references, citings, index te

**23 Word sense disambiguation using machine-readable dictionaries**

R. Krovetz, W. B. Croft

May 1989 ACM SIGIR Forum , Proceedings of the 12th annual international ACM SIGIR Forum: development in information retrieval, Volume 23 Issue 1-2

Full text available:  pdf(1.06 MB)

Additional Information: full citation, references, citing:

**24 Exploring knowledge acquisition tools for a veterinary medical expert system**

M. McLeish

June 1988 Proceedings of the first international conference on Industrial and engineering systems and expert systems - Volume 2

Full text available:  pdf(679.12 KB)

Additional Information: full citation, references

**25 Supporting compositional reuse in component-based Web engineering**

Martin Gaedke, Jörn Rehse

March 2000 Proceedings of the 2000 ACM symposium on Applied computing

Full text available:  pdf(701.38 KB)

Additional Information: full citation, references, index terms

Keywords: WebComposition, component retrieval, pattern, repository, reuse

**26 Evolutionary computing and optimization: A spanning-tree-based genetic algorithm for the rectilinear Steiner problem with obstacles**

Rita M. Hare, Bryant A. Julstrom

March 2003 Proceedings of the 2003 ACM symposium on Applied computing

Full text available:  pdf(540.11 KB)

Additional Information: full citation, abstract,

Given sets of points and obstacles in the plane, the rectilinear Steiner problem is to find a rectilinear Steiner tree---a tree made up of vertical and horizontal line segments---that has minimum total length. We consider only rectangular obstacles and further require that it is possible to connect every point to the tree via exactly one vertical and one horizontal line segment. We then describe a genetic algorithm that can find a near-optimal solution to this problem.

Keywords: Rectilinear Steiner problem, genetic algorithms, obstacles, spanning

**27 An efficient LISP-execution architecture with a new representation for list st**

Gurindar S. Sohi, Edward S. Davidson, Janak H. Patel

June 1985 ACM SIGARCH Computer Architecture News , Proceedings of the 12th an  
Computer architecture, Volume 13 Issue 3Full text available:  pdf(790.49 KB)

Additional Information: full citation, citings,

**28 Compact Storage of Binary Trees**

Paolo Sipala

July 1982 ACM Transactions on Programming Languages and Systems (TOPLAS), V

Full text available:  pdf(820.48 KB)

Additional Information: full citation, references, index terms

**29 An optimizing compiler for lexically scoped LISP**

Rodney A. Brooks, Richard P. Gabriel, Guy L. Steele

June 1982 ACM SIGPLAN Notices , Proceedings of the 1982 SIGPLAN symposium on  
6Full text available:  pdf(1.37 MB)

Additional Information: full citation, abstract, references, c

We are developing an optimizing compiler for a dialect of the LISP language. The multiprocessing supercomputer designed at Lawrence Livermore National Labor; a language primarily for symbolic processing and list manipulation, this compiler PASCAL and FORTRAN compilers for quality of compiled numerical code. The S-: signal processing ...

**30 Determinant factorization and cycle basis: encoding schemes for the repres  
incomplete graphs**

Faris N. Abuali, Roger L. Wainwright, Dale A. Schoenefeld

February 1995 Proceedings of the 1995 ACM symposium on Applied computing

Full text available:  pdf(818.45 KB)

Additional Information: full citation, references, citings, i

### 31 Papers: novel input, output, and computation: Dynamic approximation of cc constraints

Nathan Hurst, Kim Marriott, Peter Moulder

October 2002 Proceedings of the 15th annual ACM symposium on User interface s

Full text available:  pdf(397.65 KB)

Additional Information: full citation, abstract, references,

Current constraint solving techniques for interactive graphical applications cannot handle non-overlap, or containment within non-convex shapes or shapes with smooth edges. Our technique for efficiently handling such kinds of constraints based on trust region refinement. Our approach is to model these more complex constraints by a dynamically changing set of linear constraints. At each stage, these give ...

**Keywords:** constraint-solving, containment, direct manipulation, linearization of

### 32 Adaptive operator probabilities in a genetic algorithm that applies three operators

Bryant A. Julstrom

April 1997 Proceedings of the 1997 ACM symposium on Applied computing

Full text available:  pdf(489.86 KB)

Additional Information: full citation, references, index terms

**Keywords:** adaptive operator probabilities, more than two operators, rectilinear

### 33 Shallow binding in Lisp 1.5

Henry G. Baker

July 1978 Communications of the ACM, Volume 21 Issue 7

Full text available:  pdf(492.14 KB)

Additional Information: full citation, abstract, references,


Shallow binding is a scheme which allows the value of a variable to be accessed without the need for a global environment. An elegant model for shallow binding in Lisp 1.5 is presented in which context-switching is achieved by a transformation called rerooting. Rerooting is completely general and reversible, and the Lisp 1.5 interpreter will operate correctly whether or not rerooting is invoked on every evaluation.   
assoc [v, a

**Keywords:** Algol display, FUNARG's, Lisp 1.5, deep binding, environment traversal, shallow binding

### 34 Topological design of local-area networks using genetic algorithms

Reuven Elbaum, Moshe Sidi

October 1996 IEEE/ACM Transactions on Networking (TON), Volume 4 Issue 5

Full text available:  pdf(1.32 MB)

Additional Information: full citation, references, index terms

### 35 MULTI - a LISP based multiprocessing system

Donald P. McKay, Stuart C. Shapiro

August 1980

Proceedings of the 1980 ACM conference on LISP and functional pr

Full text available:  pdf(853.68 KB)

Additional Information: full citation, abstract, referen

A package of LISP functions, collectively called MULTI, which extends LISP 1.5 to defines the notion of a process within a LISP implementation using function invocation. A process is an executable entity consisting of a process template and a set of registers. The operations the process carries out. Process environments are saved in what is called a process frame. i.e. LISP frame ...

### 36 A critique of common LISP

Rodney A. Brooks, Richard P. Gabriel

August 1984

Proceedings of the 1984 ACM Symposium on LISP and functional pr

Full text available:  pdf(741.87 KB)

Additional Information: full citation, abstract, references,

A major goal of the COMMON LISP committee was to define a Lisp language with which people would be happy to stay within its confines and thus write inherently transplantable programs. The resulting language definition is too large for many short-term and medium-term projects. Some parts of COMMON LISP cannot be implemented very efficiently on stock hardware. The generality of the design with its dif ...

### 37 Seeding the population: improved performance in a genetic algorithm for the

Bryant A. Julstrom

April 1994

Proceedings of the 1994 ACM symposium on Applied computing

Full text available:  pdf(484.27 KB)

Additional Information: full citation, references, index


Keywords: combinatorial optimization, genetic algorithms, rectilinear Steiner problem

### 38 Queue-based multi-processing LISP

Richard P. Gabriel, John McCarthy

August 1984

Proceedings of the 1984 ACM Symposium on LISP and functional pr

Full text available:  pdf(1.22 MB)

Additional Information: full citation, abstract, references, ci

As the need for high-speed computers increases, the need for multi-processors increases. One of the major stumbling blocks to the development of useful multi-processors has been the lack of a language—one which is both powerful and understandable to programmers. For many programs are artificial intelligence (AI) programs, and researchers hope that the need for multi-processors is higher than ...

**39 Flow analysis and optimization of LISP-like structures**

Neil D. Jones, Steven S. Muchnick

January 1979 Proceedings of the 6th ACM SIGACT-SIGPLAN symposium on Princip

Full text available:  pdf(964.87 KB)

Additional Information: full citation, abstract, refe

In [12] the authors introduced the concept of binding time optimization and pre: methods for determining some of the binding time characteristics of programs. ] providing methods for determining the class of shapes which an unbounded data: a LISP-like program, and describe a number of uses to which that information r compilers and interpreters for ...

**40 Mathematical programming in a hybrid genetic algorithm for Steiner point p**

David J. Thunte, Pulin Sampat

February 1995 Proceedings of the 1995 ACM symposium on Applied computir

Full text available:  pdf(763.80 KB)

Additional Information: full citation, references,

Keywords: Quasi-Newton method, Steiner points, genetic algorithm, heuristic of

Results 21 - 40 of 200

Result page: previous 1 **2** 3 4

The ACM Portal is published by the Association for Computing Machinery. (

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Cont](#)Useful downloads:  Adobe Acrobat  QuickTime  Windows Med



Subscribe Register Login  
(Full Service) (Limited Service, Free)

Search: ☒ The ACM Digital Library ☐ The Guide

"genetic algorithm" and tree and lisp

Feed

Terms used genetic algorithm and tree and lisp

Sort results  
by

relevance

Save results to a Binder

Try

Search Tips

Try

☐ Open results in a new window

Display results

expanded form

Results 41 - 60 of 200

Result page: previous 1 2 **3** 4 5 6 7

Best 200 shown

#### 41 A hybrid genetic algorithm for the point to multipoint routing problem with si

Pablo Galiasso, Roger L. Wainwright

March 2001 Proceedings of the 2001 ACM symposium on Applied computing

Full text available: pdf(84.61 KB)

Additional Information: full citation, references, index term

Keywords: Steiner trees, genetic algorithm, point to multipoint routing, telecom

#### 42 Design of a LISP-based microprocessor

Guy Lewis Steele, Gerald Jay Sussman

November 1980

Communications of the ACM, Volume 23 Issue 11

Full text available: pdf(1.89 MB)

Additional Information: full citation, abstract, referer

We present a design for a class of computers whose "instruction sets" are traditional stored-program machine languages and unlike most high-level languages, data in the same way and explicitly allows programs to be manipulated as data, stored-program computer architecture. LISP differs from traditional machine languages in that it is conceptually an unordered set of ...


Keywords: LISP, SCHEME, VLSI, direct execution, garbage collection, high-level circuits, interpreters, large-scale integration, linked lists, list structure, microprocessor, recursion



**43 A simple interprocedural register allocation algorithm and its effectiveness**

Peter A. Steenkiste, John L. Hennessy

January 1989 ACM Transactions on Programming Languages and Systems (TOPLA

Full text available:  pdf(2.56 MB)


Additional Information: full citation, abstract, references, citing

Register allocation is an important optimization in many compilers, but with per- not possible to make good use of a large register set. Procedure calls limit the ir allocation, since they force variables allocated to registers to be saved and restc LISP programs due to the higher frequency of procedure calls. An interprocedur developed by simp ...

**44 A comparison of list-processing computer languages: including a detailed c 1.5, and SLIP**

Daniel G. Bobrow, Bertram Raphael

April 1964 Communications of the ACM, Volume 7 Issue 4

Full text available:  pdf(1.01 MB)

Additional Information: full citation, references, citings,

**45 N-group classification using genetic algorithms**

Aaron H. Konstam

April 1994 Proceedings of the 1994 ACM symposium on Applied computing

Full text available:  pdf(584.82 KB) Additional Information: full citation, references, citings, index terms

Keywords: classification, genetic algorithms, linear discriminant functions

**46 Using genetic algorithms to generate Steiner triple systems**

Stephen J. Hartley, Aaron H. Konstam

March 1993 Proceedings of the 1993 ACM conference on Computer science

Full text available:  pdf(748.43 KB)

Additional Information: full citation, abstract, references,

Steiner systems, particularly triple systems, are usually generated by mathemat groups and quasi-groups. When pencil-and-paper enumeration becomes infeasit to carry out exhaustive searches. This paper presents some results of using gen exhaustive search, to generate Steiner systems. A specialized mutation operato systems. Future researc ...

#### 47 An interpreter generator using tree pattern matching

Christoph M. Hoffmann, Michael J. O'Donnell

January 1979 Proceedings of the 6th ACM SIGACT-SIGPLAN symposium on Principles

Full text available:  pdf(852.20 KB)


Additional Information: full citation, abstract, refer

Equations provide a rich, intuitively understandable notation for describing nonp LISP and Lucid. In this paper, we present techniques for automatically generati to well-known techniques for generating parsers from context-free grammars. T faithful to the simple traditional mathematical meaning of the equations-no latti to explain the corr ...

#### 48 Design of an optimizing, dynamically retargetable compiler for common Lis

Rodney A. Brooks, David B. Posner, James L. McDonald, Jon L. White, Eric Benson

August 1986 Proceedings of the 1986 ACM conference on LISP and functional progr

Full text available:  pdf(1.13 MB)


Additional Information: full citation, references, citings

#### 49 Speeding up Lisp-based symbolic mathematics

Richard J. Fateman, Mark Hayden

March 1996

ACM SIGSAM Bulletin, Volume 30 Issue 1

Full text available:  pdf(1.15 MB)

Additional Information: full citation, abstract, in

Two techniques for speeding up a traditional Lisp-based symbolic manipulation s previous paper [2]. These were: using unique representations for equivalent "ke basically anything but a sum), and using hash tables for an unordered represent a complete version of Macsyma suggest that a speedup of --- in some cases --- some cases this appears t ...

#### 50 Experiments with the M & N tree-searching program

James R. Slagle, John K. Dixon

March 1970

Communications of the ACM, Volume 13 Issue 3

Full text available:  pdf(896.52 KB)

Additional Information: full citation, abstract, refere

The M & N procedure is an improvement to the mini-max backing-up procedure game-playing and other purposes. It is based on the principle that it is desirable decisions in the face of uncertainty. The mini-max procedure assigns to a MAX ( (lowest) valued successor to that node. The M & N procedure assigns to a MAX ( highest (lowest) valued succes ...

Keywords: LISP, artificial intelligence, backing-up procedures, decision theory, c min-max backing-up procedure, tree searching

**51 Programming in an Interactive Environment: the ``Lisp" Experience**

Erik Sandewall

January 1978 ACM Computing Surveys (CSUR), Volume 10 Issue 1

Full text available:  pdf(3.25 MB) Additional Information: full citation, references, citings, index terms**52 Canonical representations in Lisp and applications to computer algebra sys**


Richard J. Fateman

June 1991 Proceedings of the 1991 international symposium on Symbolic and algeb

Full text available:  pdf(1.11 MB) Additional Information: full citation, references, index terms**53 Subtree replacement systems: A unifying theory for recursive equations, LI**

Mike O'Donnell

May 1977 Proceedings of the ninth annual ACM symposium on Theory of compu

Full text available:  pdf(782.24 KB) Additional Information: full citation, abstract, referenc

Recent work on computation of functions defined by sets of recursive equations Downey and Sethi [DS76] depends on semantic interpretations of such equation approach yields similar results for a much wider class of sets of equations, includ Lucid and the combinator calculus. The application to LISP proves several conje

**54 A transputer-based parallel Lisp implementation**

M. D. Feng, C. K. Yuen

April 1992 Proceedings of the 1992 ACM annual conference on Communications

Full text available:  pdf(816.59 KB) Additional Information: full citation, references, index terms

Keywords: parallel Lisp, speculative processing, transputer, tuple space

**55 Prolog - the language and its implementation compared with Lisp**

David H D Warren, Luis M. Pereira, Fernando Pereira

August 1977 Proceedings of the 1977 symposium on Artificial intelligence and progr  
64Full text available:  pdf(670.37 KB) Additional Information: full citation, abstract, references,

Prolog is a simple but powerful programming language founded on symbolic logi a pattern matching process (&ldquo;unification&rdquo;) operating on general re logic). We briefly review the language and compare it especially with pure Lisp. techniques for implementing Prolog efficiently; in particular we describe how to matching process. These ...

**56 P-tree classification of yeast gene deletion data**

Amal Perera, Anne Denton, Pratap Kotala, William Jockheck, Willy Valdivia Granda,  
December 2002 ACM SIGKDD Explorations Newsletter, Volume 4 Issue 2

Full text available:  pdf(37.80 KB)

Additional Information: full citation, abstract, r

Genomics data has many properties that make it different from "typical" relational attributes as well as the large number of null values led us to a P-tree-based bit 1-values were counted to evaluate similarity between genes. Quantitative information was also included in the classifier. Interaction information allowed us to extend information on i ...

Keywords: P-tree, bioinformatics, data mining, genetic algorithm, genomics

**57 Evolutionary computing and optimization: Initialization is robust in evolution trees as sets of edges**

Bryant A. Julstrom, Günther R. Raidl

March 2002

Proceedings of the 2002 ACM symposium on Applied computing

Full text available:  pdf(542.67 KB)

Additional Information: full citation, abstract, referen

Evolutionary algorithms (EAs) that search spaces of spanning trees can encode a case, edge-sets for an EA's initial population should represent spanning trees of a graph that underlies the target problem instance. However, the generation of random might appear. Mechanisms based on Prim's and Kruskal's minimum spanning tree uniform mechanisms are slow, ...


Keywords: initialization, random spanning trees, sets of edges, spanning trees

**58 Experiments With Some Programs That Search Game Trees**

James R. Slagle, John E. Dixon

April 1969

Journal of the ACM (JACM), Volume 16 Issue 2

Full text available:  pdf(1.14 MB)


Additional Information: full citation, abstract, references, ci

Many problems in artificial intelligence involve the searching of large trees of alternating game-playing and theorem-proving. The problem of efficiently searching large trees "dynamic ordering" is described, and the older minimax and Alpha comparison purposes. Performance figures are given for six variations of the game "depth ratio" is de ...

**59 Is it a tree, a DAG, or a cyclic graph? A shape analysis for heap-directed pointers**

Rakesh Ghiya, Laurie J. Hendren

January 1996 Proceedings of the 23rd ACM SIGPLAN-SIGACT symposium on Principles

Full text available:  pdf(1.51 MB)

Additional Information: full citation, references, citations, inde

**60 A bidirectional data driven Lisp engine for the direct execution of Lisp in pa**

C. K. Yuen, W. F. Wong

June 1989 ACM SIGARCH Computer Architecture News, Volume 17 Issue 4

Full text available:  pdf(761.13 KB)




Additional Information: full citation, index terms

Results 41 - 60 of 200

Result page: previous 1 2 **3** 4

The ACM Portal is published by the Association for Computing Machinery. (

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Cont](#)

Useful downloads:  Adobe Acrobat  QuickTime  Windows Med

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE



Membership Publications/Services Standards Conferences Careers/Jobs

**IEEE Xplore<sup>®</sup>**  
RELEASE 1.6Welcome  
United States Patent and Trademark Office[Help](#) [FAQ](#) [Terms](#) [IEEE Peer Review](#) [Quick Links](#)

» Search

Welcome to IEEE Xplore<sup>®</sup>

- ☐ Home
- ☐ What Can I Access?
- ☐ Log-out

## Tables of Contents

- ☐ Journals & Magazines
- ☐ Conference Proceedings
- ☐ Standards

## Search

- ☐ By Author
- ☐ Basic
- ☐ Advanced

## Member Services

- ☐ Join IEEE
- ☐ Establish IEEE Web Account
- ☐ Access the IEEE Member Digital Library

Your search matched **2** of **1006282** documents.A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance** in **Descending** order.**Refine This Search:**

You may refine your search by editing the current search expression or entering new one in the text box.

☐ Check to search within this result set**Results Key:****JNL** = Journal or Magazine   **CNF** = Conference   **STD** = Standard**1 Two fast tree-creation algorithms for genetic programming***Luke, S.;*

Evolutionary Computation, IEEE Transactions on , Volume: 4 , Issue: 3 , Sept. 2000

Pages: 274 - 283

[\[Abstract\]](#)   [\[PDF Full-Text \(300 KB\)\]](#)   **IEEE JNL****2 Genetic programming of fuzzy logic production rules***Edmonds, A.N.; Burkhardt, D.; Adjei, O.;*

Evolutionary Computation, 1995., IEEE International Conference on , Volume: 2 , 29 Nov.-1 Dec. 1995

Pages: 765 - 770 vol.2

[\[Abstract\]](#)   [\[PDF Full-Text \(340 KB\)\]](#)   **IEEE CNF**

[Home](#) | [Log-out](#) | [Journals](#) | [Conference Proceedings](#) | [Standards](#) | [Search by Author](#) | [Basic Search](#) | [Advanced Search](#) | [Join IEEE](#) | [Web Account](#) | [New this week](#) | [OPAC Linking Information](#) | [Your Feedback](#) | [Technical Support](#) | [Email Alerting](#) | [No Robots Please](#) | [Release Notes](#) | [IEEE Online Publications](#) | [Help](#) | [FAQ](#) | [Terms](#) | [Back to Top](#)

Copyright © 2004 IEEE — All rights reserved